Trinadh Phani Ginjupalli (002117493)

Program Structures and Algorithms (Fall 2021)

Assignment 5 - Parallel sort

Link to the assignment’s repository - <https://github.com/Phani56/INFO6205-Assignments>

**Tasks Performed**

* Added functionality in ParSort.java to run on multiple threads using ForkJoinPool.
* Updated Main.java to vary cutoff, arraysize and thread count from command line.
* Analysed *Cutoff vs Thread count* by altering the thread count from 1 to 1024 from multiple cutoff values.
* Analysed *Cutoff vs Array size* for array sizes from 2 million to 7 million increasing by a count of 1mil.
* Saved the results for above runs in files in the project. Path: *src/results/parallelsort/*
* Plotted graphs for both the cases to infer a relationship

Link to both the analysis here - [Parallel sort Analysis](https://docs.google.com/spreadsheets/d/1d-j3Eo9FxEIhhb2PJD6FjPgllFf7gt2tyQbU_DHmzEw/edit?usp=sharing). This contains sheets with

1. Cutoff vs Threads
2. Cutoff vs ArraySize
3. Multiple sheets with the results from each individual run

**Conclusion**

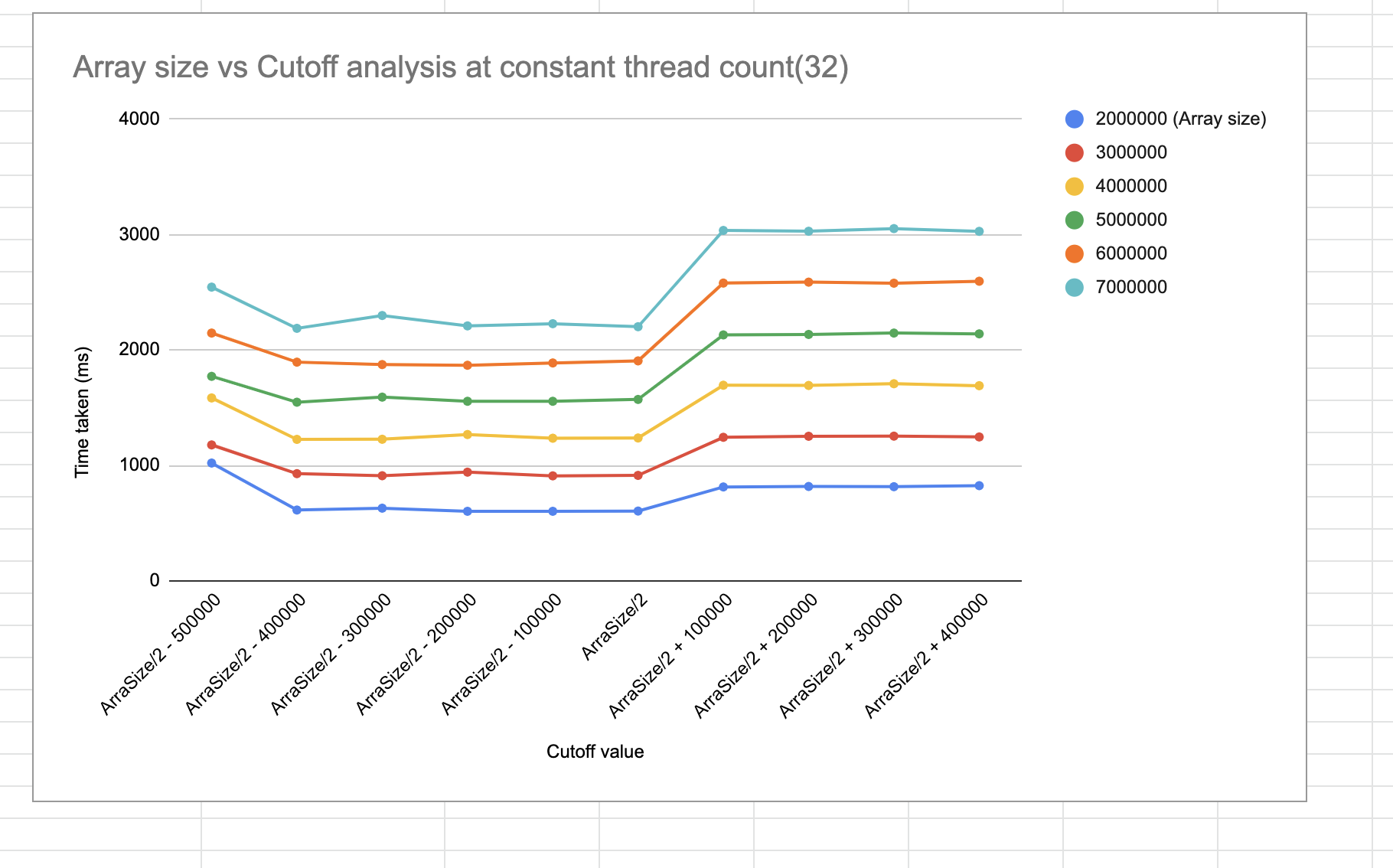
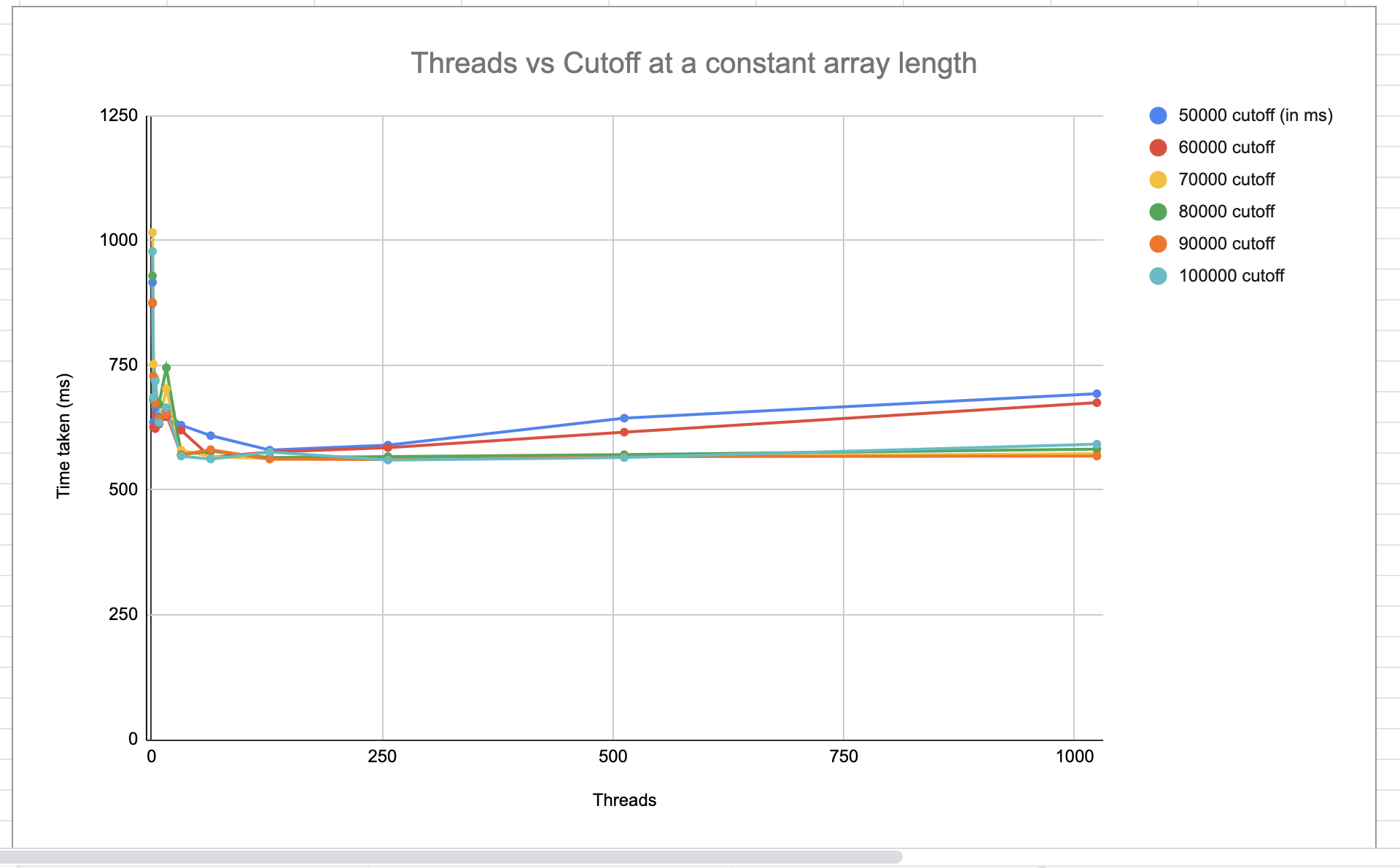
For Cutoff vs Threads I have varied the thread count from 1 to 1024 with array size constant(2000000).

From the data of multiple runs shown below can see that the run time decreases as the thread count reaches a low at around **32** threads and then slowly takes an upward turn.

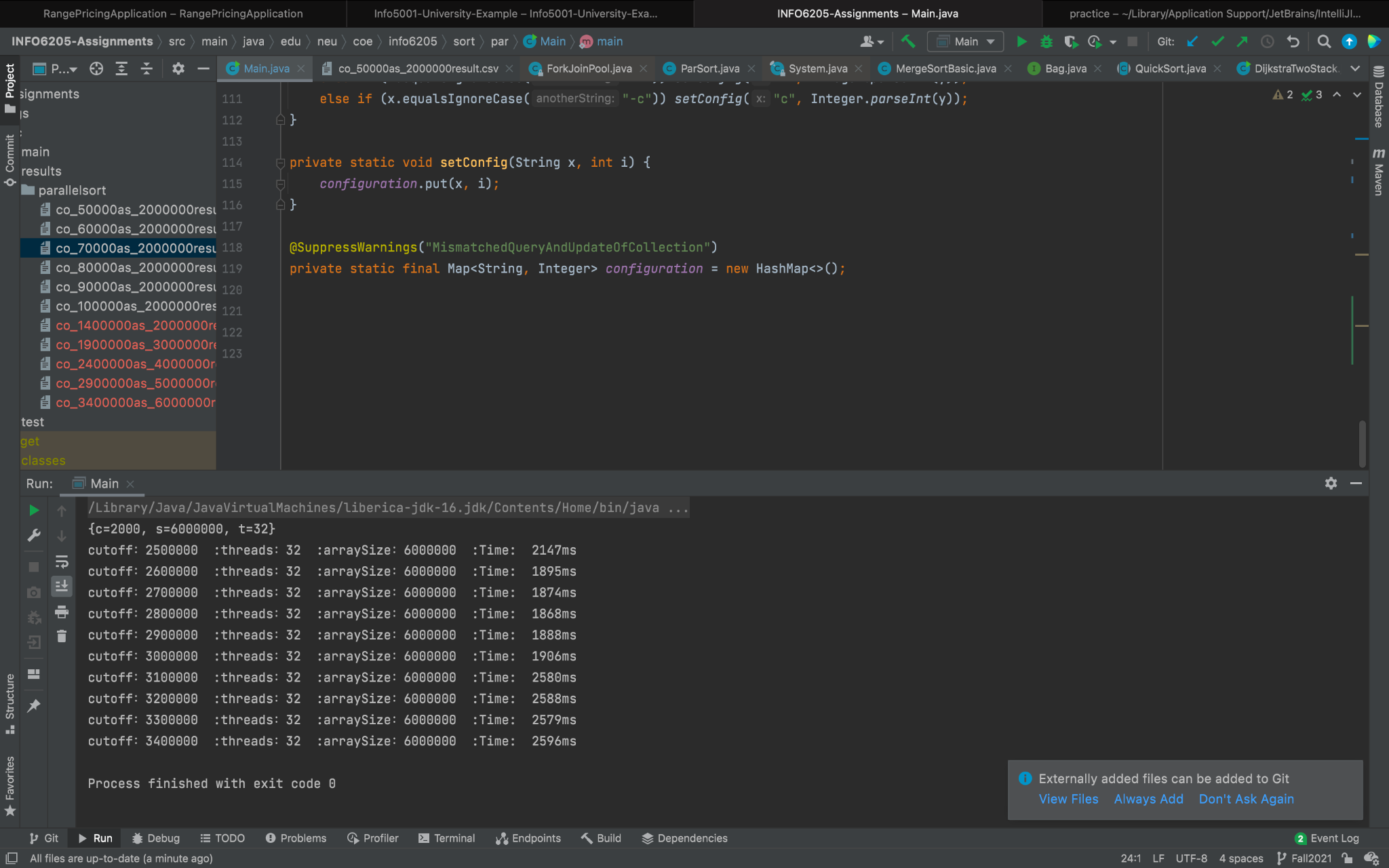
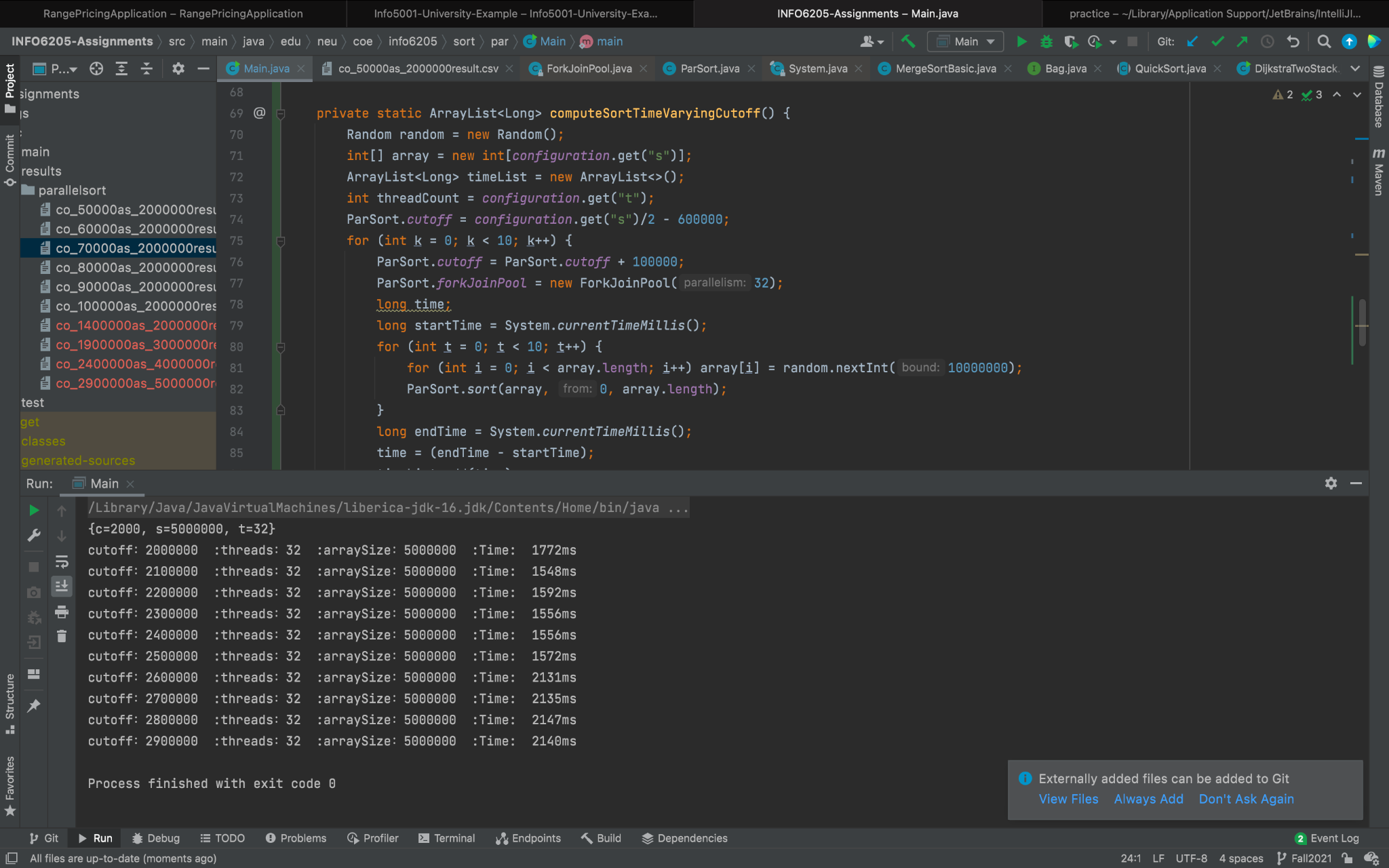
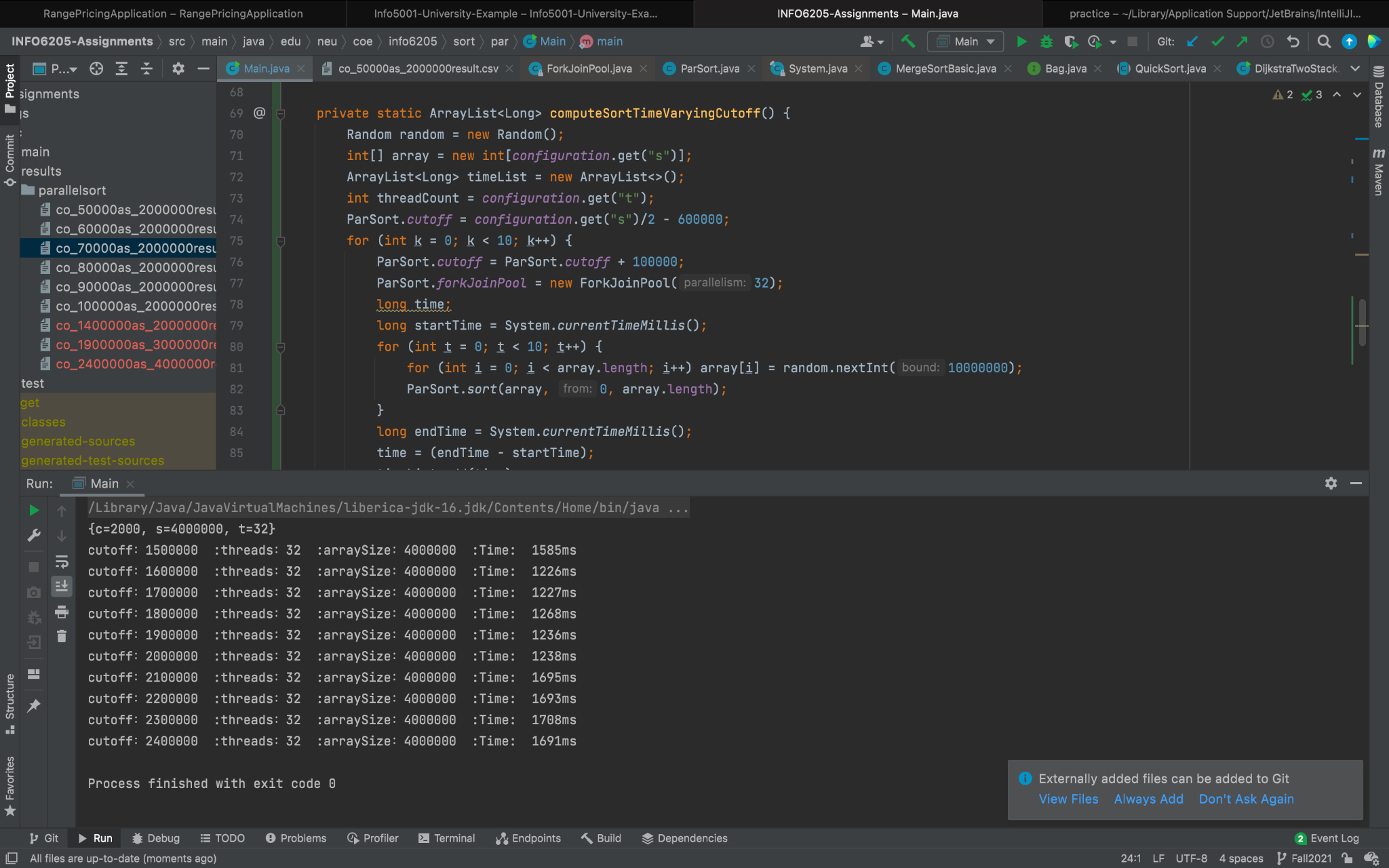
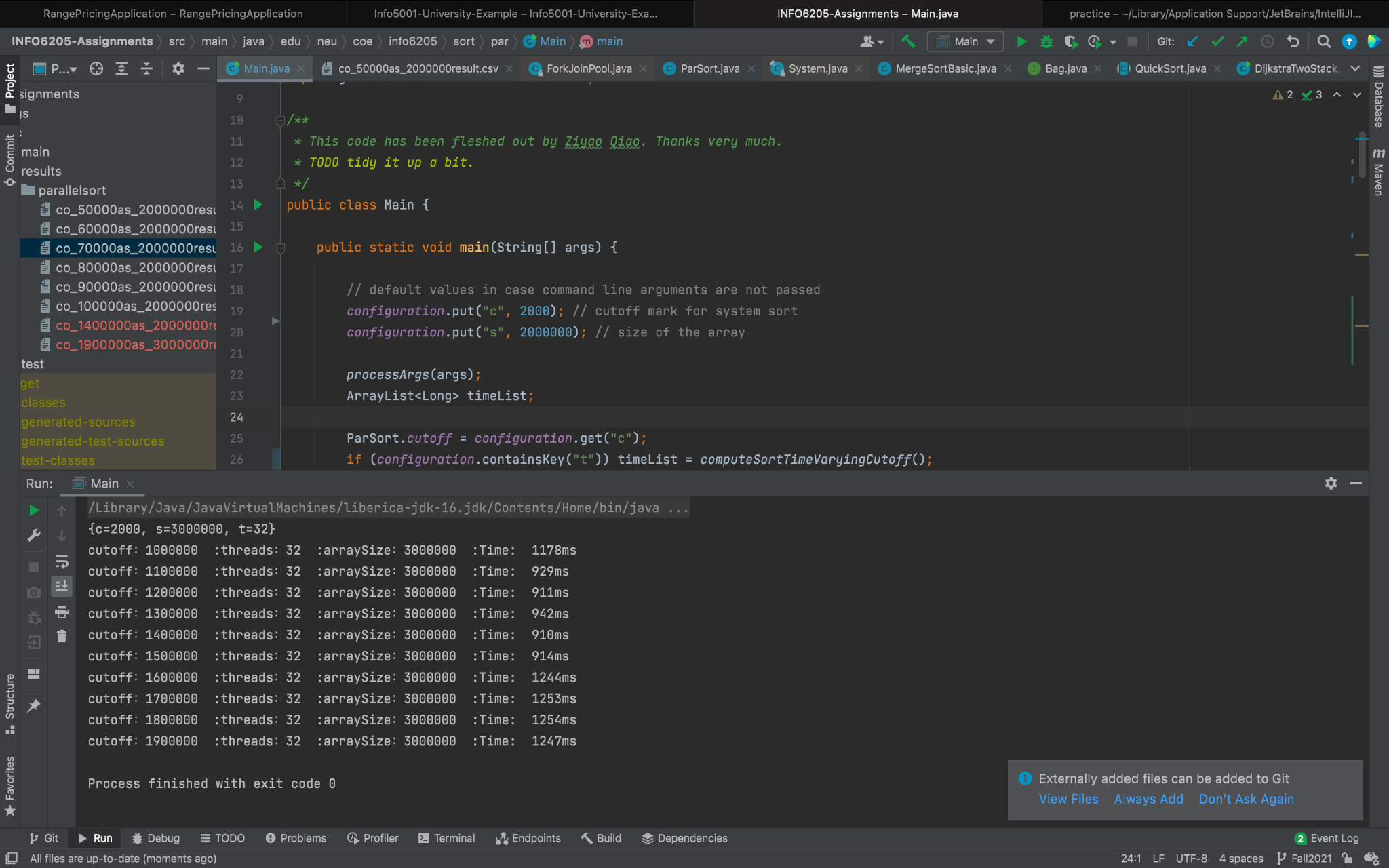
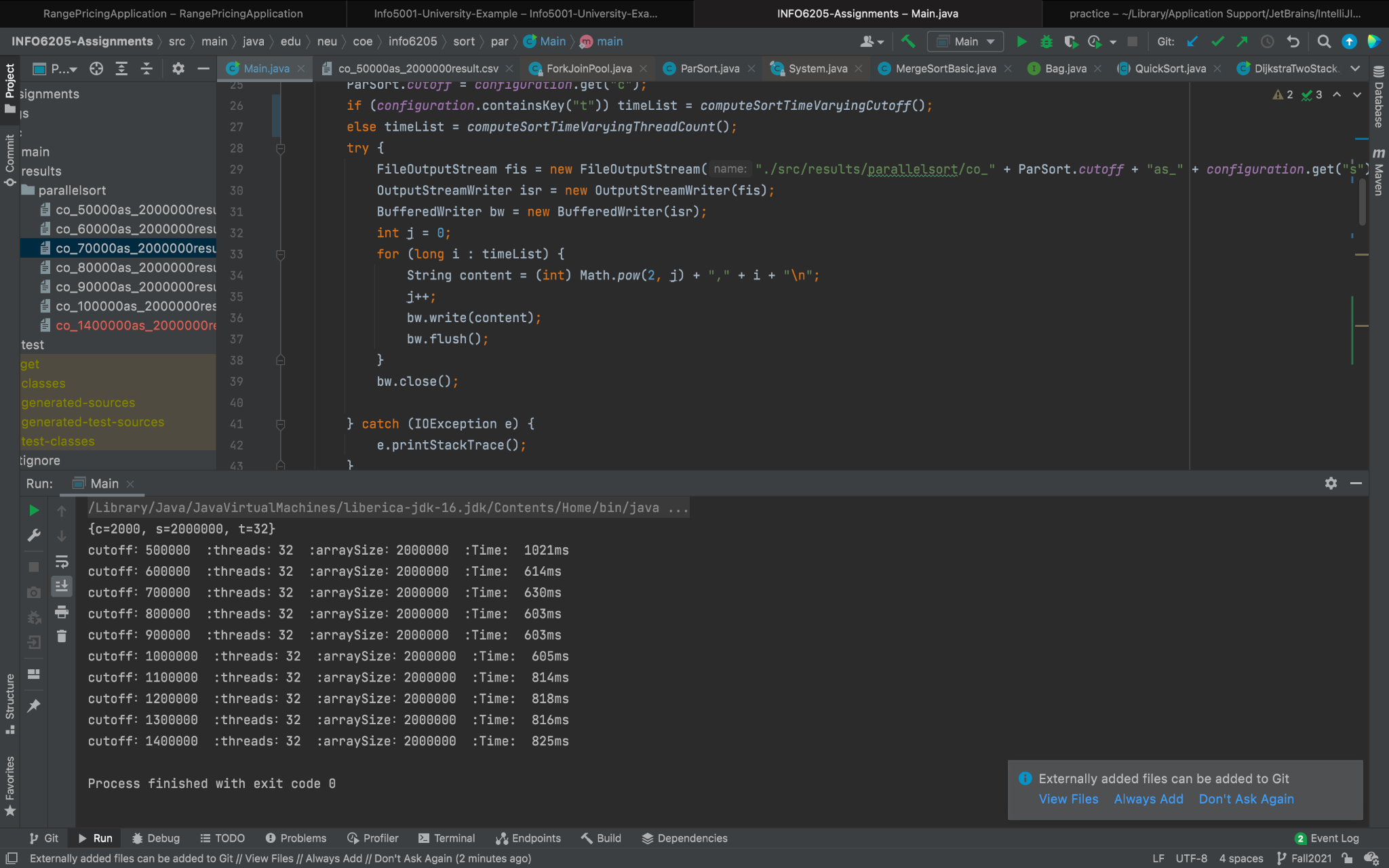
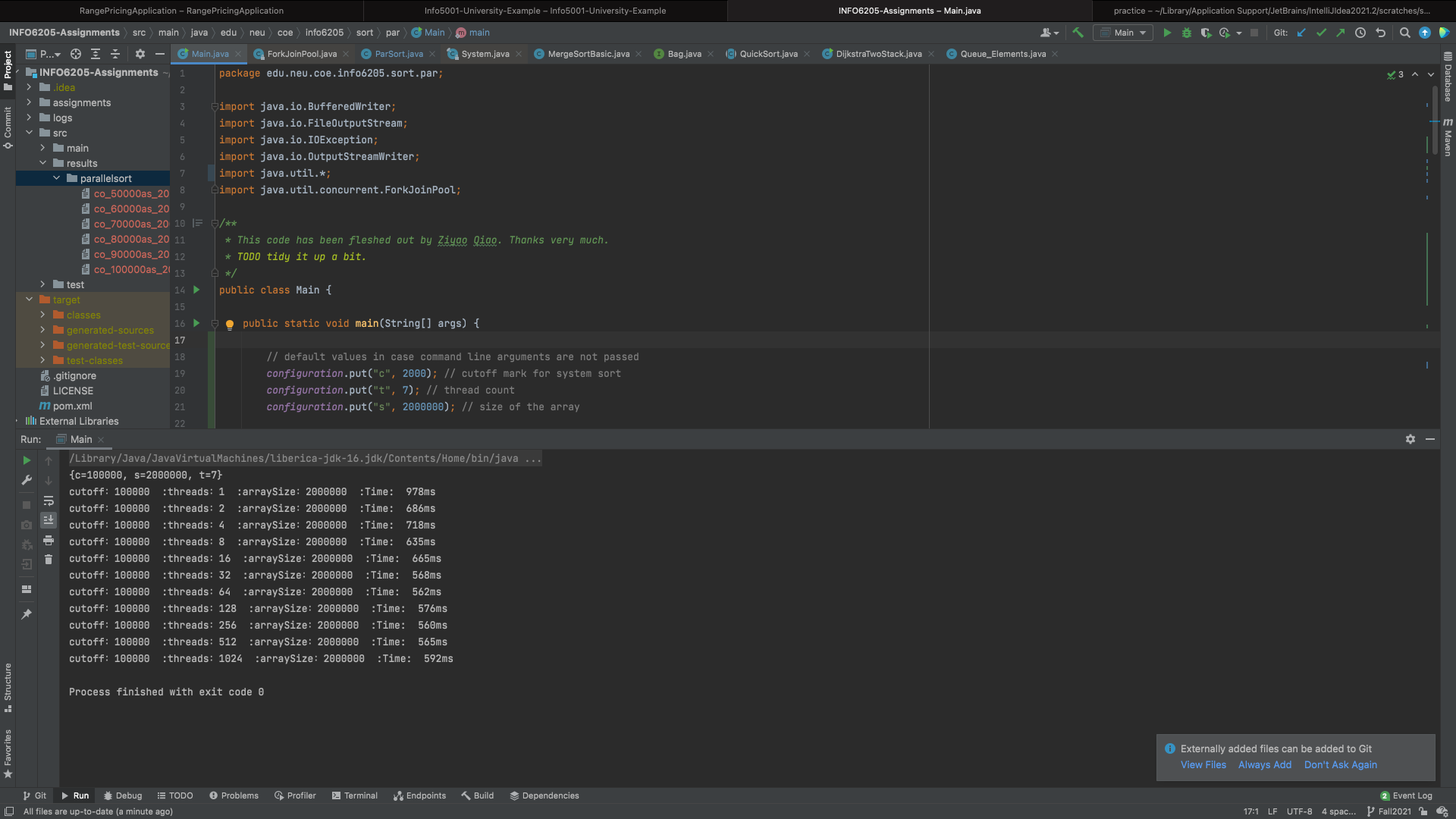
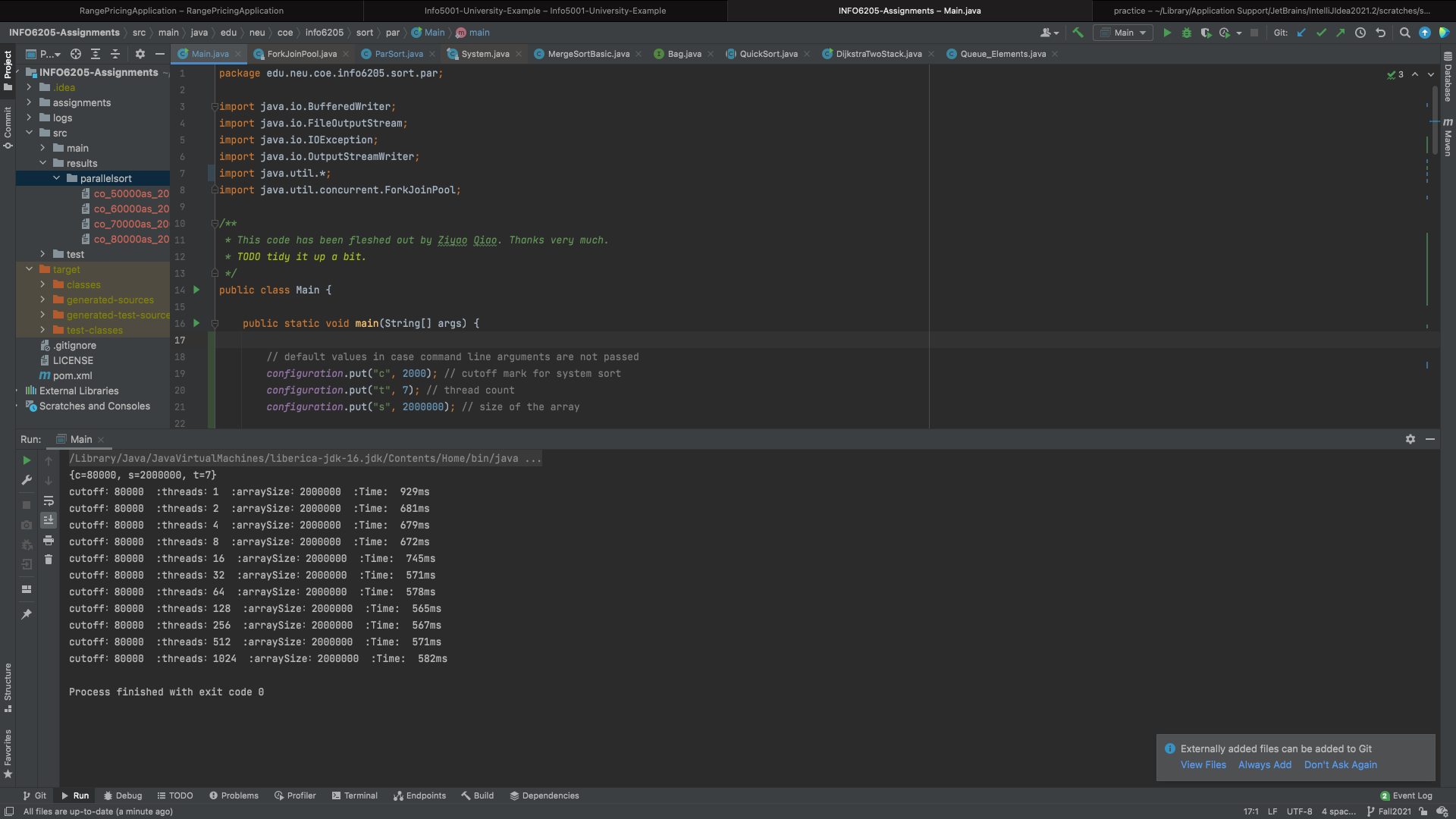
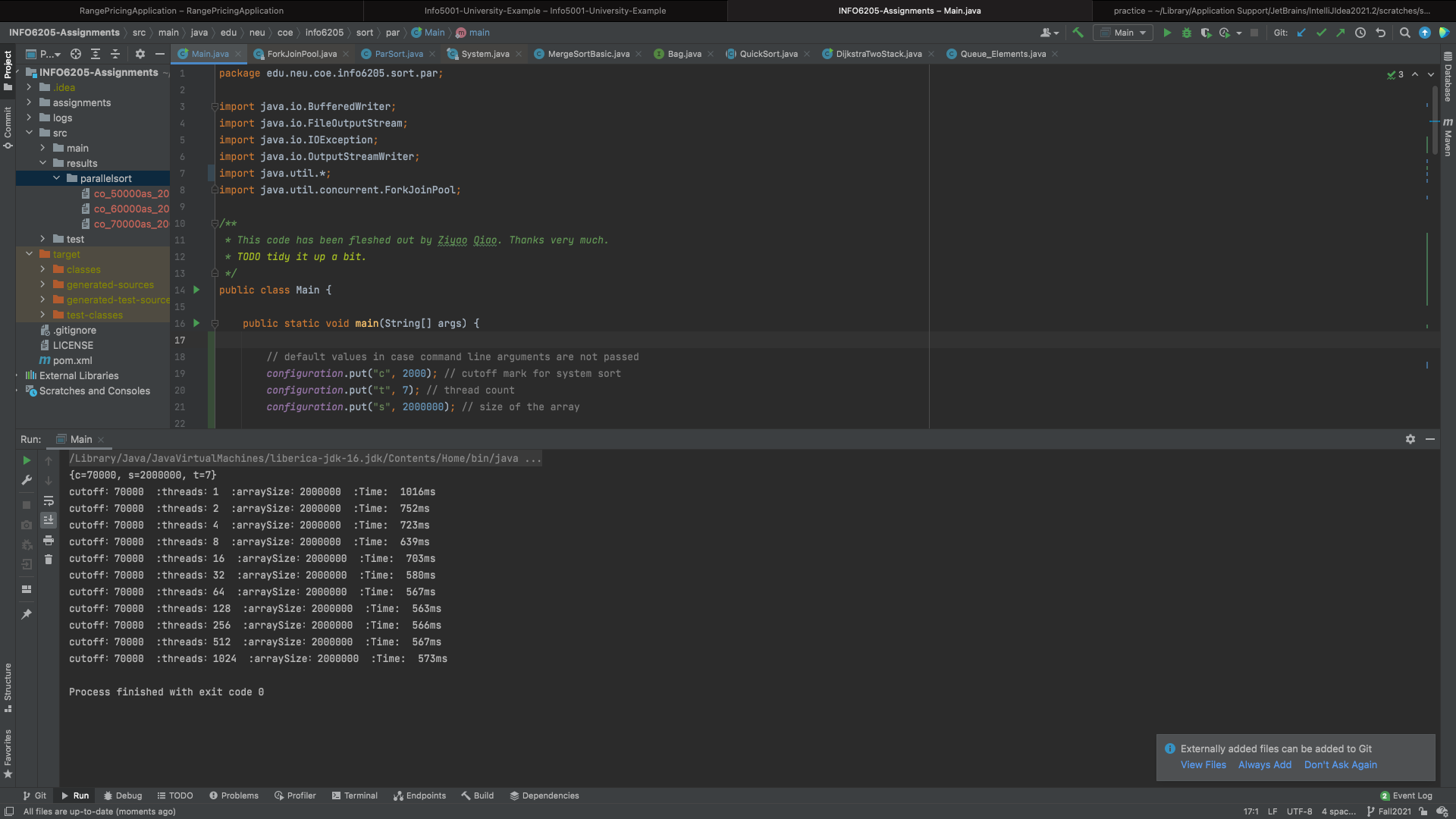
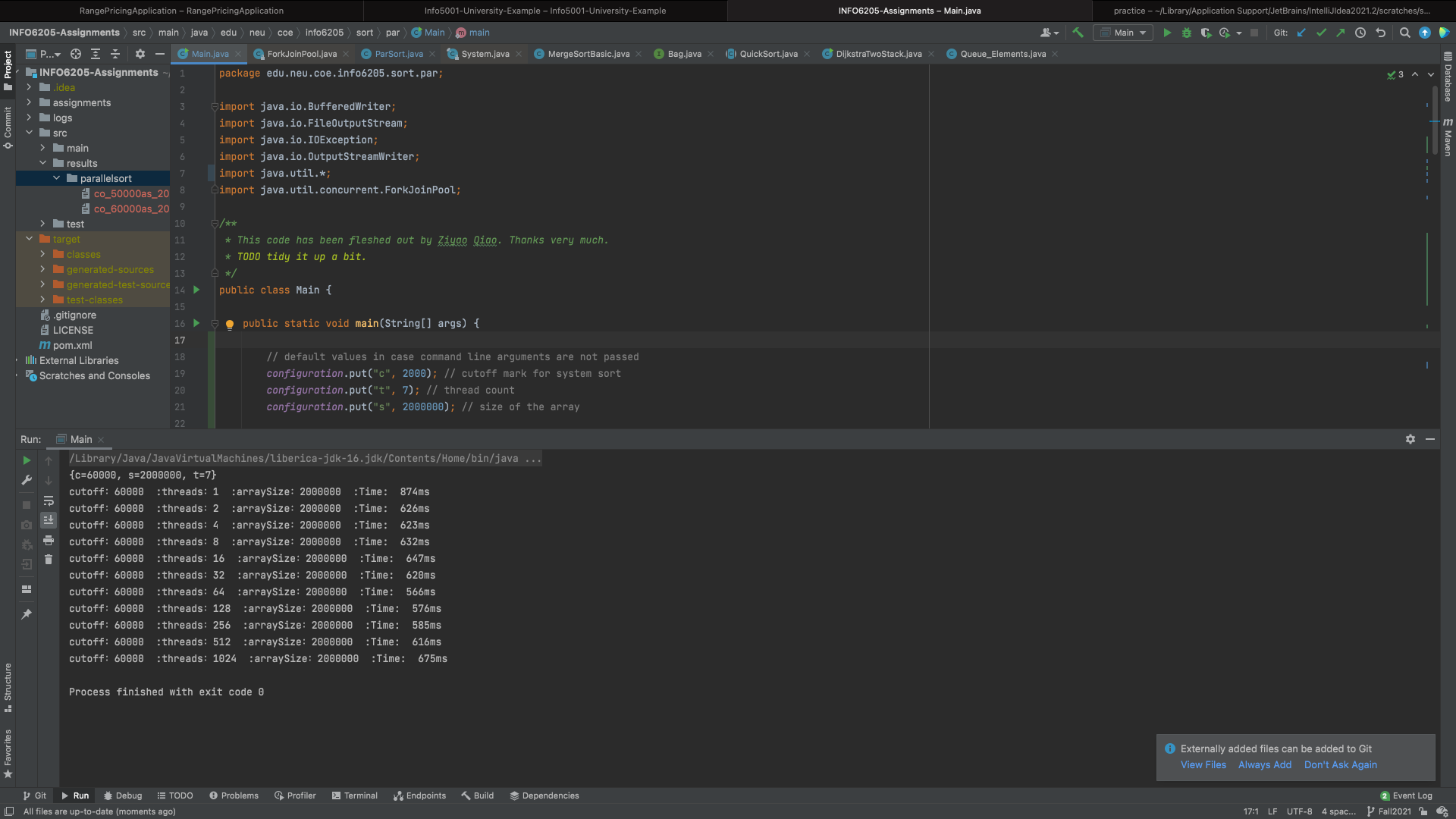
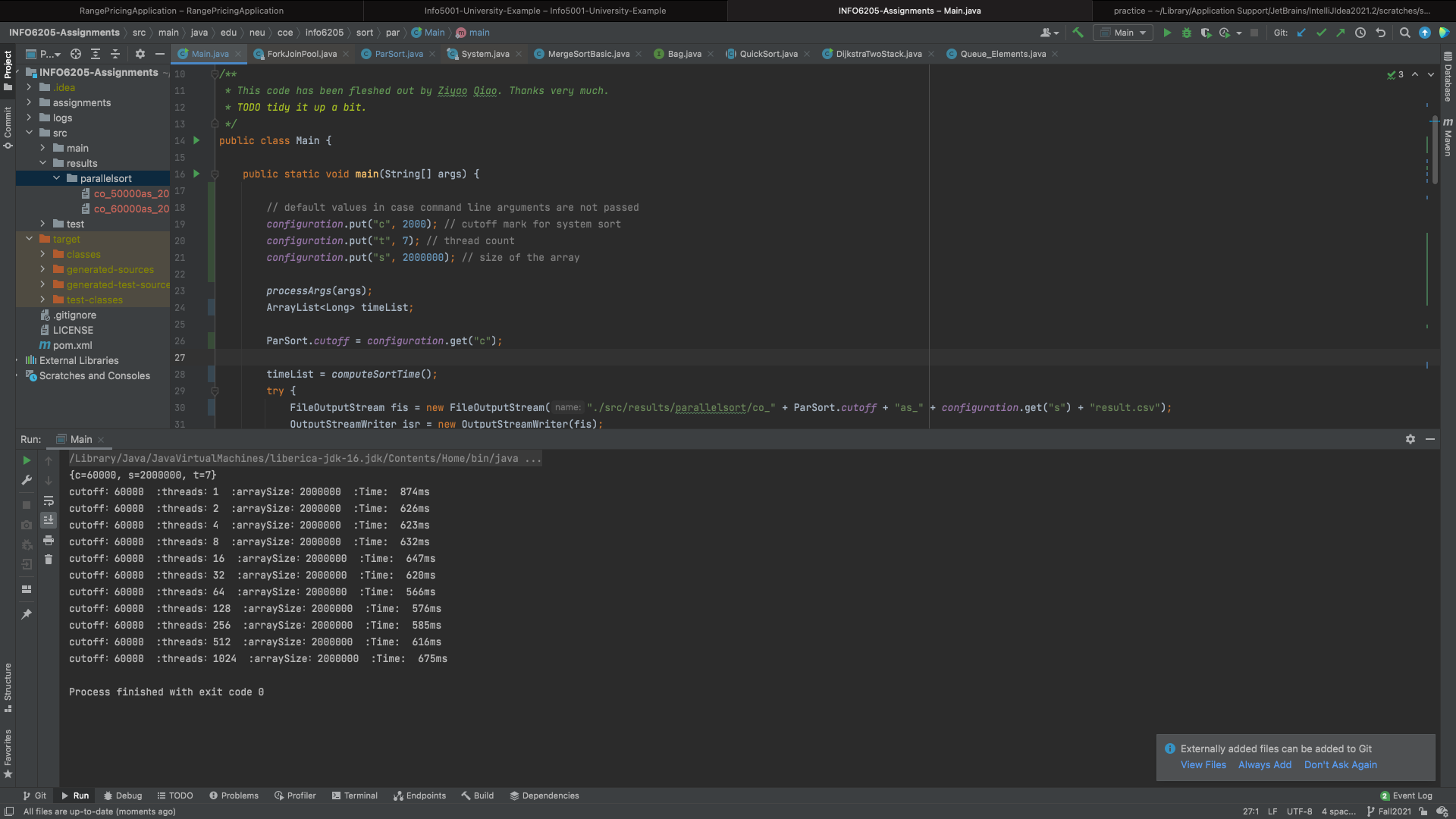
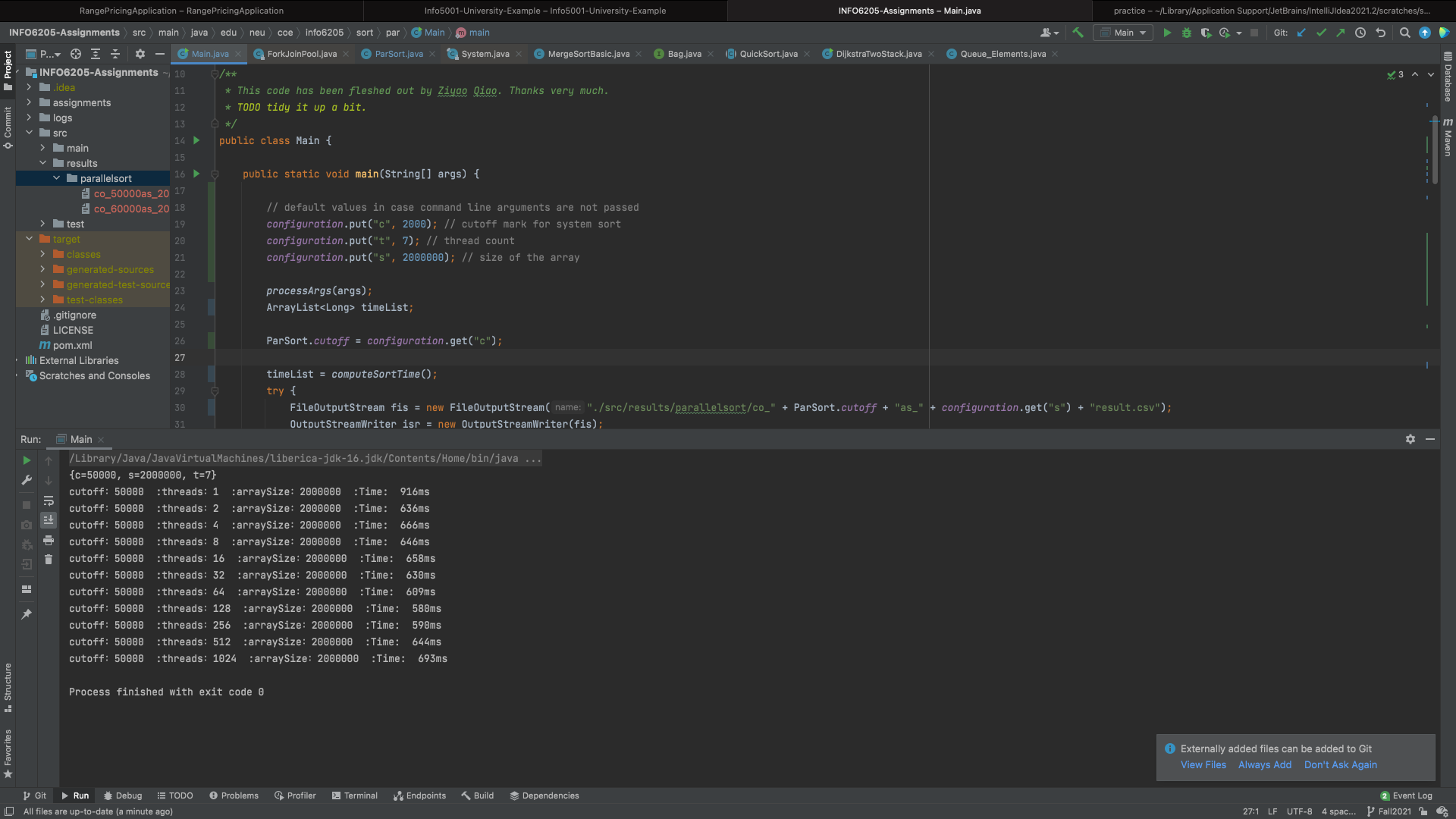
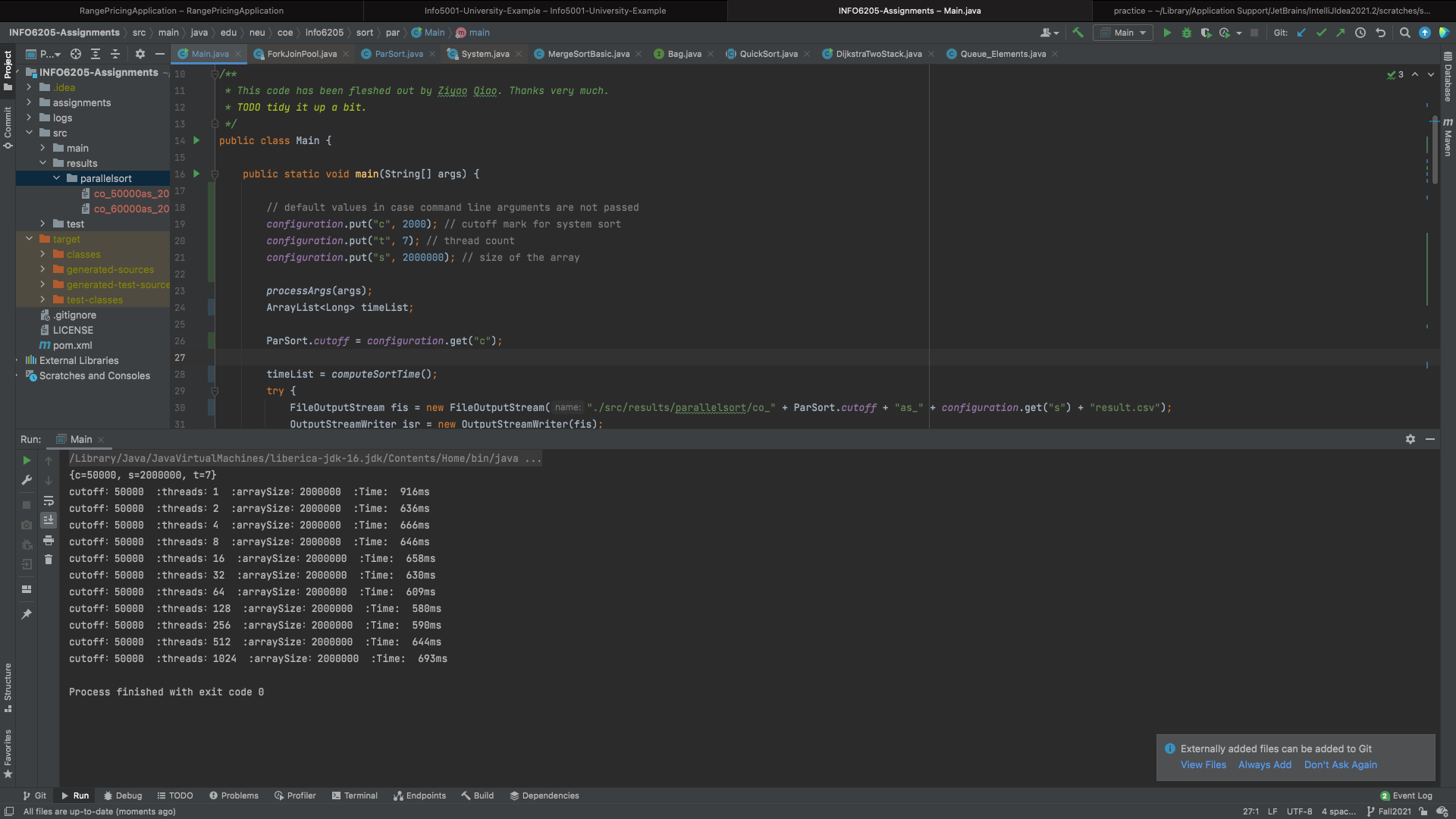
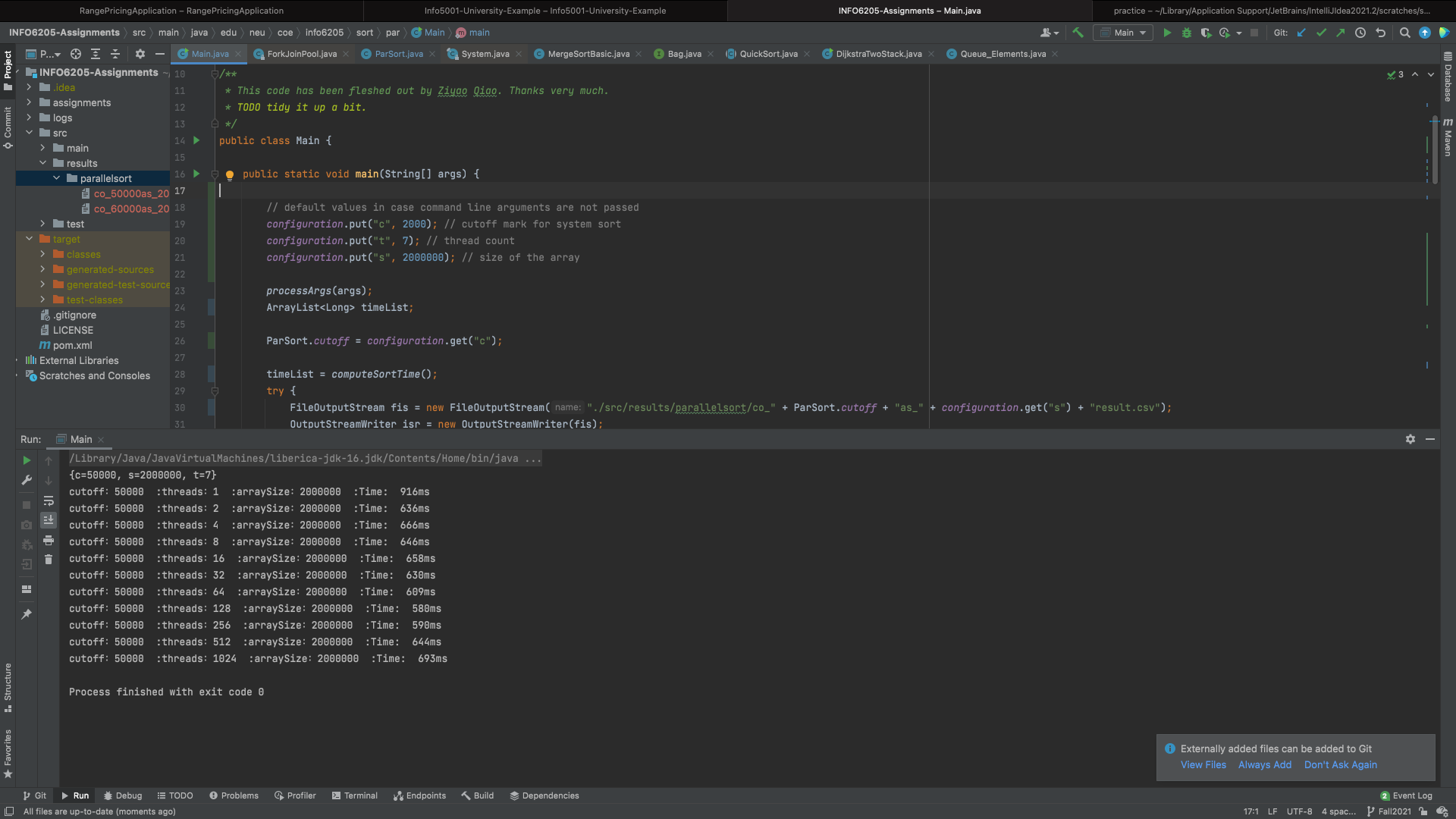
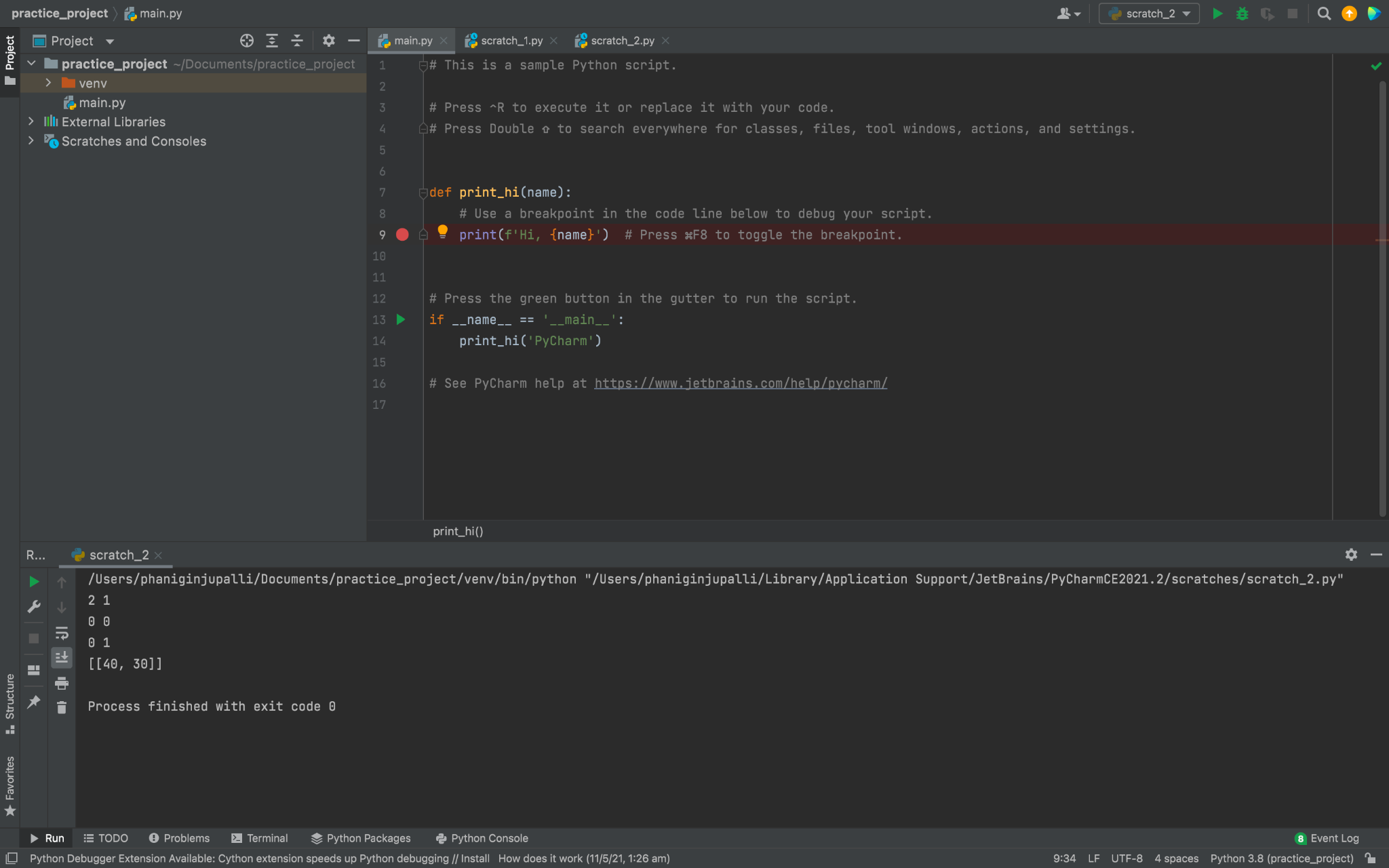
For Cutoff vs ArraySize

I ran this case with a constant thread count of 32. The run time was almost at a constant rate till the cutoff was half of the array size and then it sharply increased at the midpoint to maintain the constant rate again.

My conclusion is to **use degree of parallelism as 32 and cutoff to system sort when it reaches half of the array size.**



**Screenshots for all the runs**

****